Amendments to the Claims:

The following listing of claims will replace all prior versions, and listings, of claims in the application:

1. (Currently Amended) ~~An implementation unit~~A distributed actuation allocation system, comprising:

a plurality of implementation units grouped into a plurality of modules comprising at least one of the plurality of implementation units or a plurality of implementation unit sub-modules, wherein each of the plurality of implementation unit sub-modules includes at least one of the plurality of implementation units~~,~~;

a controller that generates a system goal; and

~~an~~at least one allocator that receives the system goal, partitions the system goal into a plurality of sub-goals, one for each of the plurality of modules, based on an allocation parameter~~, and allocates the sub-goals to at least one of:~~

~~the plurality of implementation units, and~~

~~a plurality of implementation modules comprising a group of the plurality of implementation units~~,

wherein ~~the allocator controls the implementation of the plurality of implementation units~~the at least one allocator allocates at least one sub-goal to at least one of the plurality of sub-modules if the corresponding module includes a plurality of sub-modules, and

the at least one allocator actuates the at least one of the plurality of implementation units within the at least one sub-module based on the sub-goal if the corresponding module does not include the plurality of sub-modules.

2. (Original) The system, as recited in claim 1, further comprising:

at least one sensor that detects an implementation state of at least one of the plurality of implementation units and provides the controller with the implementation state,

wherein the controller generates the system goal based on the implementation state.

3. (Original) The system, as recited in claim 1, wherein at least one of the plurality of implementation units is an air jet.

4. (Original) The system as recited in claim 1, further comprising:

an input source that provides system objective to the controller,

wherein the controller generates the system goal based on the system objective signal.

5. (Original) The system, as recited in claim 4, wherein the input source is a computer.

6. (Original) The system, as recited in claim 1, wherein the controller is remotely located from the plurality of actuators.

7. (Original) The system, as recited in claim 1, further comprising:

at least one sensor that detects an implementation state of at least one of the plurality of implementation units and provides the controller with the implementation state; and

an input source that provides an input signal to the controller,

wherein the controller generates the system goal based on the implementation state and the input signal.

8. (Original) The system, as recited in claim 1, wherein the allocation parameter is predefined.

9. (Original) The system, as recited in claim 1, wherein the allocation parameter is identity of the plurality of implementation modules under the control of the allocator.

10. (Original) The system, as recited in claim 1, wherein the allocator partitions the system goal into the plurality of sub-goals based on the allocation parameter and a second allocation parameter,

wherein the second allocation parameter is a location identifier for at least one of:

the plurality of implementation units, and

the implementation modules.

11. (Original) The system, as recited in claim 1, wherein the allocator partitions the system goal into the plurality of sub-goals based on the allocation parameter, a second allocation parameter and a third allocation parameter,

wherein the second allocation parameter is a location identifier and the third allocation parameter is a weighting factor for at least one of:

the plurality of implementation units, and

the implementation modules.

12. (Original) The system, as recited in claim 1, wherein the allocator further includes a plurality of hierarchical allocation levels each of which include at least one module allocator that allocate the sub-goals.

13. (Currently Amended) A computer-implemented method for allocating a system instruction to a plurality of actuators, comprising:

grouping the plurality of actuators into a plurality of ~~module actuators~~actuator modules comprising ~~at least one of a plurality of sub-module actuators and~~ at least one of the

plurality of actuators or a plurality of actuator sub-modules, wherein each of the plurality of actuator sub-modules ~~actuators~~ includes at least one of the plurality of actuators;

partitioning the system instruction into a plurality of sub-instructions, one for each of the plurality of actuator modules, ~~actuators~~ based on at least one allocation parameter;

allocating each of the plurality of sub-instructions to at least one of the plurality of ~~sub-module actuators~~actuator sub-modules if the corresponding actuator module ~~actuator~~ includes the plurality of actuator sub-modules ~~actuators~~; and

actuating the at least one of the plurality of actuators within the at least one actuator sub-module ~~actuator~~ based on the sub-instructions, if the corresponding actuator module ~~actuator~~ does not include the plurality of actuator sub-modules ~~actuators~~.

14. (Previously Presented) The method, as recited in claim 13, wherein the grouping step is based on a physical layout of the plurality of actuators.

15. (Currently Amended) ~~The method, as recited in claim 13,~~ A method for allocating a system instruction to a plurality of actuators, comprising:

grouping the plurality of actuators into a plurality of actuator modules comprising at least one of the plurality of actuators or a plurality of actuator sub-modules, wherein each of the plurality of actuator sub-modules includes at least one of the plurality of actuators;

partitioning the system instruction into a plurality of sub-instructions for each of the plurality of actuator modules based on at least one allocation parameter;

allocating each of the plurality of sub-instructions to at least one of the plurality of actuator sub-modules if the corresponding actuator module includes the plurality of actuator sub-modules; and

actuating the at least one of the plurality of actuators within the at least one actuator sub-module based on the sub-instructions, if the corresponding actuator module does not include the plurality of actuator sub-modules,

further comprising:

partitioning each of the plurality of sub-instructions into a plurality of second sub-instructions for each of the ~~sub-module actuators~~actuator sub-modules;

determining if each of the plurality of ~~sub-module actuators~~actuator sub-modules includes at least one second actuator sub-module ~~actuator~~;

allocating each of the plurality of second sub-instructions to the least one second actuator sub-module ~~actuators~~ for each of the ~~sub-module actuators~~actuator sub-modules determined to include the at least one second actuator sub-module ~~actuator~~; and

actuating at least one of the plurality of actuators disposed within the sub-module based on the second sub-instructions for each of the actuator sub-modules ~~actuators~~ determined not to include the at least one second actuator sub-module ~~actuator~~.

16. (Currently Amended) The method, as recited in claim 13, wherein the allocation parameter includes at least one of:

the number of the plurality of ~~sub-module actuators~~actuator sub-modules; and

the number of the plurality of ~~module actuators~~actuator modules.

17-27. (Canceled)

28. (New) The system, as recited in claim 1, wherein at least one sub-module includes a plurality of second sub-modules, and the at least one allocator:

partitions at least one sub-goal into a plurality of second sub-goals, one for each of the second sub-modules,

determines if the sub-modules include at least one second sub-module,

allocates one of the second sub-goals to each of the sub-modules determined to include the at least one second sub-module, and

actuating at least one of the plurality of implementation units disposed within the sub-module based on the second sub-goals for each of the sub-modules determined not to include the at least one second sub-module.